Improved Complexity Results on k-Coloring P_t-Free Graphs

Shenwei Huang

School of Computing Science
Simon Fraser University, Burnaby B.C., V5A 1S6, Canada
shenweih@sfu.ca

Abstract. A graph is H-free if it does not contain an induced subgraph isomorphic to H. We denote by P_k and C_k the path and the cycle on k vertices, respectively. In this paper, we prove that 4-COLORING is NP-complete for P_7-free graphs, and that 5-COLORING is NP-complete for P_6-free graphs. These two results improve two previously best results and almost complete the classification of complexity of k-COLORING P_t-free graphs for $k \geq 4$ and $t \geq 1$, leaving as the only missing case 4-COLORING P_6-free graphs. We expect that 4-COLORING is polynomial time solvable for P_6-free graphs; in support of this, we describe a polynomial time algorithm for 4-COLORING P_6-free graphs which are also C_4-free.

1 Introduction

We consider computational complexity issues related to vertex coloring problems restricted to P_k-free graphs. It is well known that the usual k-COLORING problem is NP-complete for any fixed $k \geq 3$. Therefore, there has been considerable interest in studying its complexity when restricted to certain graph classes. One of the most remarkable results in this respect is that k-COLORING is polynomially solvable for perfect graphs. More information on this classical result and related work on coloring problems restricted to graph classes can be found in several surveys, e.g, [22,23].

We continue the study of k-COLORING problem for P_t-free graphs. This problem has been given wide attention in recent years and much progress has been made through substantial efforts by different groups of researchers [4,5,6,9,13,15,17,18,19,21,24]. We summarize these results and explain our new results below.

We refer to [3] for standard graph theory terminology and [11] for terminology on computational complexity. Let $G = (V, E)$ be a graph and $\mathcal{H}$ be a set of graphs. We say that G is *$\mathcal{H}$-free* if G does not contain any graph $H \in \mathcal{H}$ as an induced subgraph. In particular, if $\mathcal{H} = \{H\}$ or $\mathcal{H} = \{H_1, H_2\}$, we simply say that G is H-free or (H_1, H_2)-free. Given any positive integer t, let P_t and C_t be the path and cycle on t vertices, respectively. A *linear forest* is a disjoint

union of paths. We denote by $G + H$ the disjoint union of two graphs G and H. We denote the complement of G by $\bar{G}$. The neighborhood of a vertex x in G is denoted by $N_G(x)$, or simply $N(x)$ if the context is clear. Given a vertex subset $S \subseteq V$ we denote by $N_S(x)$ the neighborhood of x in S, i.e., $N_S(x) = N(x) \cap S$. For two disjoint vertex subsets X and Y we say that X is *complete,* respectively *anti-complete*, to Y if every vertex in X is adjacent, respectively non-adjacent, to every vertex in Y. The *girth* of a graph G is the length of the shortest cycle.

A *k-coloring* of a graph $G = (V, E)$ is a mapping $\phi : V \rightarrow \{1, 2, \ldots, k\}$ such that $\phi(u) \neq \phi(v)$ whenever $uv \in E$. The value $\phi(u)$ is usually referred to as the *color* of u under ϕ. We say G is *k-colorable* if G has a k-coloring. The problem k-COLORING asks if an input graph admits an k-coloring. The k-LIST COLORING problem asks if an input graph G with lists $L(v) \subseteq \{1, 2, \ldots, k\}$, $v \in V(G)$, has a coloring ϕ that *respects* the lists, i.e., $\phi(v) \in L(v)$ for each $v \in V(G)$.

In the *pre-coloring extension of k-coloring* we assume that (a possible empty) subset $W \subseteq V$ of G is pre-colored with $\phi_W : W \rightarrow \{1, 2, \ldots, k\}$ and the question is whether we can extend ϕ_W to a k-coloring of G. We denote the problem of pre-coloring extension of k-coloring by k^*-COLORING. Note that k-COLORING is a special case of k^*-COLORING, which in turn is a special case of k-LIST COLORING.

Kamiński and Lozin [17] showed that, for any fixed $k \geq 3$, the k-COLORING problem is NP-complete for the class of graphs of girth at least g for any fixed $g \geq 3$. Their result has the following immediate consequence.

Theorem 1 ([17]). *For any $k \geq 3$, the k-COLORING problem is NP-complete for the class of H-free graphs whenever H contains a cycle.*

Holyer [16] showed that 3-COLORING is NP-complete for line graphs. Later, Leven and Galil [20] extended this result by showing that k-COLORING is also NP-complete for line graphs for $k \geq 4$. Because line graphs are claw-free, these two results together have the following consequence.

Theorem 2 ([16,20]). *For any $k \geq 3$, the k-COLORING problem is NP-complete for the class of H-free graphs whenever H is a forest with a vertex of degree at least 3.*

Due to Theorems 1 and 2, only the case in which H is a linear forest remains. In this paper we focus on the case where H is a path. The k-COLORING problem is trivial for P_t-free graphs when $t \leq 3$. The first non-trivial case is P_4-free graphs. It is well known that P_4-free graphs (also called *cographs*) are perfect and therefore can be colored optimally in polynomial time by Grötschel et al. [14]. Alternatively, one can color cographs using the *cotree representation* of a cograph, see, e.g., [22]. Hoàng et al. [15] developed an elegant recursive algorithm

showing that the k-COLORING problem can be solved in polynomial time for P_5-free graphs for any fixed k.

Woeginger and Sgall [24] proved that 5-COLORING is NP-complete for P_8-free graphs and 4-COLORING is NP-complete for P_{12}-free graphs. Later, Le et al. [19] proved that 4-COLORING is NP-complete for P_9-free graphs. The sharpest results so far are due to Broersma et al. [4,6].

Theorem 3 ([6]). *4-COLORING is NP-complete for P_8-free graphs and 4^*-COLORING is NP-complete for P_7-free graphs.*

Theorem 4 ([4]). *6-COLORING is NP-complete for P_7-free graphs and 5^*-COLORING is NP-complete for P_6-free graphs.*

In this paper we strengthen these NP-completeness results. We prove that 5-COLORING is NP-complete for P_6-free graphs and that 4-COLORING is NP-complete for P_7-free graphs. We shall develop a novel general framework of reduction and prove both results simultaneously in Section 2. This leaves the k-COLORING problem for P_t-free graphs unsolved only for $k = 4$ and $t = 6$, except for 3-COLORING. (The complexity status of 3-COLORING P_t-free graphs for $t \geq 7$ is open. It is even unknown whether there exists a fixed integer $t \geq 7$ such that 3-COLORING P_t-free graphs is NP-complete.) We will focus on the case $k = 4$ and $t = 6$. In Section 3, we shall explain why the framework established in Section 2 is not sufficient to prove the NP-completeness of 4-COLORING for P_6-free graphs. However, we were able to develop a polynomial time algorithm for 4-COLORING (P_6, C_4)-free graphs. These two results suggest that 4-COLORING might be polynomially solvable for P_6-free graphs. Finally, we give some related remarks in Section 4.

2 The NP-completeness Results

We begin this section by pointing out an error in the proof of NP-completeness of 6-COLORING P_7-free graphs [4]. In this paragraph we follow the notation of Broersma et al. [4]. They used a reduction from 3-SAT to the problem of 6-COLORING for P_7-free graphs. In [4], the authors constructed a graph G_I for an arbitrary instance I of 3-SAT in such a way that I is satisfiable if and only if G_I is 6-colorable. Furthermore, they claimed that G_I is P_7-free. Unfortunately, the last claim is not true in general. Here is one counterexample. Suppose I is an instance of 3-SAT which contains only one clause $C_1 = x_1 \vee \bar{x_2} \vee x_3$. Then $\bar{x_1} y_1 b_{11} d_1 b_{13} y_3 \bar{x_3}$ is an induced P_7 in the graph G_I from [4].

Next we shall prove our main results.

Theorem 5. *5-COLORING is NP-complete for P_6-free graphs.*

Theorem 6. *4-COLORING is NP-complete for P_7-free graphs.*

Instead of giving two independent proofs for Theorems 5 and 6, we provide a unified framework. The *chromatic number* of a graph G, denoted by $\chi(G)$, is the minimum positive integer k such that G is k-colorable. The *clique number* of a graph G, denoted by $\omega(G)$, is the maximum size of a clique in G. A graph G is called *k-critical* if $\chi(G) = k$ and $\chi(G - v) < k$ for any vertex v in G. We call a k-critical graph *nice* if G contains three independent vertices $\{c_1, c_2, c_3\}$ such that $\omega(G - \{c_1, c_2, c_3\}) = \omega(G) = k - 1$.

Let I be any 3-SAT instance with variables $X = \{x_1, x_2, \ldots, x_n\}$ and clauses $\mathcal{C} = \{C_1, C_2, \ldots, C_m\}$, and let H be a nice k-critical graph with three specified independent vertices $\{c_1, c_2, c_3\}$. We construct the graph G_I as follows.

- Introduce for each variable x_i a *variable component* T_i which is isomorphic to K_2, labeled by $x_i\bar{x_i}$. Call these vertices *X-type*.

- Introduce for each variable x_i a vertex d_i. Call these vertices *D-type*.

- Introduce for each clause $C_j = y_{i_1} \vee y_{i_2} \vee y_{i_3}$ a *clause component* H_j which is isomorphic to H, where y_{i_t} is either x_{i_t} or $\bar{x_{i_t}}$. Denoted three specified independent vertices in H_j by c_{i_tj} for $t = 1, 2, 3$. Call c_{i_tj} *C-type* and all remaining vertices *U-type*.

For any C-type vertex c_{ij} we call x_i or $\bar{x_i}$ its *corresponding literal vertex*, depending on whether $x_i \in C_j$ or $\bar{x_i} \in C_j$.

- Connect each U-type vertex to each D-type and X-type vertices.

- Connect each C-type vertex c_{ij} to d_i and its corresponding literal vertex.

Lemma 1. *Let H be a nice k-critical graph. Suppose G_I is the graph constructed from H and a 3-SAT instance I. Then I is satisfiable if and only if G_I is $(k+1)$-colorable.*

Proof. We first assume that I is satisfiable and let σ be a truth assignment satisfying each clause C_j. Then we define a mapping $\phi : V(G) \to \{1, 2, \ldots, k+1\}$ as follows.

- Let $\phi(d_i) := k + 1$ for each i.

- If $\sigma(x_i)$ is TRUE, then $\phi(x_i) := k + 1$ and $\phi(\bar{x_i}) := k$. Otherwise, let $\phi(x_i) =: k$ and $\phi(\bar{x_i}) =: k + 1$.

- Let $C_j = y_{i_1} \vee y_{i_2} \vee y_{i_3}$ be any clause in I. Since σ satisfies C_j, at least one literal in C_j, say y_{i_t} $(t \in \{1, 2, 3\})$, is TRUE. Then the corresponding literal vertex of c_{i_tj} receives the same color as d_{i_t}. Therefore, we are allowed to color c_{i_tj} with color k. In other words, we let $\phi(c_{i_tj}) := k$.

- Since $H_j = H$ is k-critical, $H_j - c_{i_tj}$ has a $(k-1)$-coloring $\phi_j : V(H_j - c_{i_tj}) \to \{1, 2, \ldots, k - 1\}$. Let $\phi =: \phi_j$ on $H_j - c_{i_tj}$.

It is easy to check that ϕ is indeed a $(k+1)$-coloring of G_I.

Conversely, suppose ϕ is a $(k+1)$-coloring of G_I. Since $H_1 = H$ is a nice k-critical graph, the largest clique of U-type vertices in H_1 has size $k-1$. Let R_1 be such a clique. Note that $\omega(G_I) = k+1$ and $R = R_1 \cup T_1$ is a clique of size $k+1$. Therefore, any two vertices in R receive different colors in any $(k+1)$-coloring of G_I. Without loss of generality, we may assume $\{\phi(x_1), \phi(\bar{x_1})\} = \{k, k+1\}$. Because every U-type vertex is adjacent to every X-type and D-type vertex, we have the following three properties of ϕ.

(P1) $\{\phi(x_i), \phi(\bar{x_i})\} = \{k, k+1\}$ for each i.

(P2) $\phi(d_i) \in \{k, k+1\}$ for each i.

(P3) $\phi(u) \in \{1, 2, \ldots, k-1\}$ for each U-type vertex.

Next we construct a truth assignment σ as follows.

- Set $\sigma(x_i)$ to be TRUE if $\phi(x_i) = \phi(d_i)$ and FALSE otherwise.

It follows from (P1) and (P2) that σ is a truth assignment. Suppose σ does not satisfy $C_j = y_{i_1} \vee y_{i_2} \vee y_{i_3}$. Equivalently, $\sigma(y_{i_t})$ is FALSE for each $t = 1, 2, 3$. It follows from our definition of σ that the corresponding literal vertex of $c_{i_t j}$ receives different color from the color of d_{i_t} under ϕ. Hence, $\phi(c_{i_t j}) \notin \{k, k+1\}$ for $t = 1, 2, 3$ and this implies that ϕ is a $(k-1)$-coloring of $H_j = H$ by (P3). This contradicts the fact that $\chi(H) = k$. $\square$

Lemma 2. *Let H be a nice k-critical graph. Suppose G_I is the graph constructed from H and a 3-SAT instance I. If H is P_t-free where $t \geq 6$, then G_I is P_t-free as well.*

Proof. Suppose $P = P_t$ is an induced path with $t \geq 6$ in G_I. We first prove the following claim.

Claim A. *P contains no U-type vertex.*

Proof of Claim A. Suppose that u is a U-type vertex on P that lies in some clause component H_j. For any vertex x on P we denote by x^- and x^+ the left and right neighbor of x on P, respectively. Let us first consider the case when u is the left endvertex of P. If u^+ belongs to H_j, then $P \subseteq H_j$, since u is adjacent to all X-type and D-type vertices and P is induced. This contradicts the fact that H is P_t-free. Hence, u^+ is either X-type or D-type. Note that u^{++} must be C-type or U-type. In the former case we conclude that u^{+++} is U-type since C-type vertices are independent. Hence, $|P| \leq 3$ and this is a contradiction. In the latter case we have $|P| \leq 4$ for the same reason. Note that $|P| = 4$ only if P follows the pattern $U(X \cup D)UC$, namely the first vertex of P is U-type, the second vertex of P is X-type or D-type, and so on. Next we consider the case that u has two neighbors on P.

Case 1. Both u^- and u^+ belong to H_j. In this case $P \subseteq H_j$ and this contradicts the fact that $H = H_j$ is P_t-free.

Case 2. $u^- \in H_j$ but $u^+ \notin H_j$. Then u^+ is either X-type or D-type. Since each U-type vertex is adjacent to each X-type and D-type vertex, u^- is a C-type vertex and hence it is an endvertex of P. Now $|P| \leq 2 + 4 - 1 = 5$.

Case 3. Neither u^- nor u^+ belongs to H_j. Now both u^- and u^+ are X-type or D-type. Since each U-type vertex is adjacent to each X-type and D-type vertex, $P \cap U = \{u\}$ and $P \cap (X \cup D) = \{u^-, u^+\}$. Hence, $|P| \leq 5$. ($|P| = 5$ only if P follows the pattern $C(X \cup D)U(X \cup D)C$). □

Let C_i (resp. $\bar{C}_i$) be the set of C-type vertices that connect to x_i (resp. $\bar{x_i}$). Let $G_i = G[\{T_i \cup \{d_i\} \cup C_i \cup \bar{C}_i\}]$. Note that $G - U$ is disjoint union of G_i, $i = 1, 2, \ldots, n$. By Claim A, $P \subseteq G_i$ for some i. Let P' be a sub-path of P of order 6 . Since $C_i \cup \bar{C}_i$ is independent, $|P' \cap (C_i \cup \bar{C}_i)| \leq 3$. Hence, $|P' \cap (C_i \cup \bar{C}_i)| = 3$ and thus $\{d_i, x_i, \bar{x_i}\} \subseteq P'$. This contradicts the fact that P' is induced since d_i has three C-type neighbors on P'. □

Due to Lemmas 1 and 2, the following theorem follows.

Theorem 7. *Let $t \geq 6$ be an fixed integer. Then k-COLORING is NP-complete for P_t-free graphs whenever there exists a P_t-free nice $(k-1)$-critical graph.* □

Proof of Theorems 5 *and* 6. Let H_1 be the graph as follows. H_1 has vertex set

$$V(H_1) = \{u, u_1, u_2, u_3, c_1, c_2, c_3\},$$

and edge set

$$E(H) = \{c_1u_3, c_2u_3, c_1u_2, c_3u_2, c_2u_1, c_3u_1, u_1u_2, u_2u_3, u_3u_1, uc_1, uc_2, uc_3\}.$$

Let $H_2 = C_7$ be the 7-cycle. It is easy to check that H_1 is a P_6-free nice 4-critical graph and that H_2 is a P_7-free nice 3-critical graph. Applying Theorem 7 with $H = H_i$ $(i = 1, 2)$ will complete our proof. □

3 The Polynomial Result

Having proved Theorems 5 and 6 the next question now is whether 4-COLORING is NP-complete for P_6-free graphs. Unfortunately, the framework established in Section 2 is not sufficient to prove the NP-completeness. In fact, there is no P_6-free nice 3-critical graph. Suppose H is P_6-free nice 3-critical graphs with $\{c_1, c_2, c_3\}$ being independent. Note that H is not perfect by the definition of nice critical graph. So H contains an induced C_t or $\bar{C}_t$ for some odd integer $t \geq 5$ by the Strong Perfect Graph Theorem [8]. Since $\chi(H) = 3$ and H is P_6-free, H must contain an induced $C = C_5$. Now all three c_i's have to belong to C since H is 3-critical. This is impossible since C_5 contains at most two independent vertices.

This negative result suggests that 4-COLORING P_6-free graphs might be solved in polynomial time. But it seems difficult to prove this since the usual techniques for 3-COLORING (see, e.g., [4,6,21,24]) do not apply. It turns out that the problem becomes easier if we forbid one more induced subgraph, namely a cycle. For example, if we consider P_6-free graphs that are also triangle-free, then the 4-COLORING problem becomes trivial since every triangle-free P_6-free graph is 4-colorable (see, e.g., [22]).

Next we shall prove 4-COLORING is polynomially solvable for (P_6, C_4)-free graphs. The coloring algorithm given below makes use of the following well-known lemma.

Lemma 3 ([10]). *Let G be a graph in which every vertex has a list of colors of size at most 2. Then checking whether G has a coloring respecting these lists is solvable in polynomial time.*

Randerath and Schiermeyer [21] proved that one can decide in polynomial time that if a P_6-free graph is 3-colorable. Very recently Broersma et al. [4] proved that the same applies to list coloring.

Lemma 4 ([4]). *3-LIST COLORING can be solved in polynomial time for P_6-free graphs.*

Now we are ready to prove our main result in this section.

Theorem 8. *4-COLORING is polynomially solvable for (P_6, C_4)-free graphs.*

Proof. Let G be a (P_6, C_4)-free graph. It is easy to see that G is k-colorable if and only if every block of G is k-colorable. In addition, all blocks of G can be found in linear time using depth first search. So we may assume that G is 2-connected. Also, we assume that G is K_5-free. Otherwise G is not 4-colorable; and it takes $O(n^5)$ time to detect such a K_5. Let us first assume that G contains an induced C_5, and let $C = v_0v_1v_2v_3v_4v_0$ be such an induced C_5.

We call a vertex $v \in V \setminus C$ an *i-vertex* if $|N(v) \cap C| = i$. Let S_i be the set of i-vertices where $0 \leq i \leq 5$. Note that $G = V(C) \cup \bigcup_{i=0}^{5} S_i$. By the C_4-freeness of G we have the following simple facts.

- S_5 must be a clique and $S_4 = \emptyset$.
- If $v \in S_3$, then $N_C(v)$ induces a P_3 and if $v \in S_2$, then $N_C(v)$ induces a P_2.

In the following all indices are modulo 5. Let $S_3(v_i)$ be the set of 3-vertices whose neighborhood on C is $\{v_i, v_{i-1}, v_{i+1}\}$ and $S_2(v_i)$ be the set of 2-vertices whose neighborhood on C is $\{v_{i-2}, v_{i+2}\}$. We also define $S_1(v_i)$ to be the set of 1-vertices that has v_i as their unique neighbor on C. Clearly, $S_p = \bigcup_{i=0}^{4} S_p(v_i)$ for $p = 1, 2, 3$. Note that $|S_5| \leq 1$ otherwise G is not 4-colorable. Further, it follows from the C_4-freeness of G that $S_3(v_i)$ is a clique for each i. So $|S_3(v_i)| \leq 2$ since

G is K_5-free. Hence, $|C \cup S_5 \cup S_3| \leq 16$ and there are at most 4^{16} different 4-colorings of $C \cup S_5 \cup S_3$. Clearly, G is 4-colorable if and only if there exists at least one such coloring that can be extended to G. Therefore, it suffices to explain how to decide if a given 4-coloring ϕ of $C \cup S_5 \cup S_3$ can be extended to a 4-coloring of G in polynomial time. Equivalently, we want to decide in polynomial time if G admits a 4-list coloring with input lists as follows.

$$L(v) = \begin{cases} \{1,2,3,4\} & \text{if } v \notin C \cup S_5 \cup S_3. \\ \phi(v) & \text{otherwise} \end{cases}$$

We say that vertices with list size 1 have been *pre-colored*. Now we *update* the graph as follows. For any pre-colored vertex v and any $x \in N(v)$ we remove color $\phi(v)$ from the list of x, i.e., let $L(x) := L(x) \setminus \{\phi(v)\}$. It is easy to see that $|L(x)| \leq 2$ for any $x \in S_2$ after updating the graph. Next we consider 0-vertices.

Claim B. *S_0 is anti-complete to $S_1 \cup S_2$. In addition, any two 0-vertices that lie in the same component of S_0 have exactly same neighbors in S_3.*

Proof of Claim B. The first claim follows directly from the P_6-freeness of G. To prove the second claim it suffices to show that $N_{S_3}(x) = N_{S_3}(y)$ holds for any edge $xy \in E$ in S_0. By contradiction assume that there exists an edge xy in S_0 such that x has a neighbor z in S_3 with $yz \notin E$. Without loss of generality, we assume $z \in S_3(v_0)$. Then $yxzv_1v_2v_3$ would induce a P_6 in G. □

Let A be an arbitrary component of S_0. Since G is 2-connected, A has at least one neighbor, say x, in S_3. By Claim B, $\phi(x)$ does not appear in the lists of vertices in A at all. So, we can decide if ϕ can be extended to A in polynomial time by Lemma 4. Since S_0 has at most n components, it takes polynomial time to check if ϕ can be extended to S_0.

Now we consider 1-vertices. Our goal is to branch on a subset of vertices in either S_1 or S_2 in such a way that after branching the vertices in S_1 that are not pre-colored are anti-complete to the vertices in S_2 that are not pre-colored. We want to accomplish such branching with only polynomial cost. If we do achieve that then we can decide in polynomial time if ϕ can be extended to S_1 and S_2 (independently) by applying Lemmas 4 and 3, respectively. Therefore, in the following we focus on branching procedure and refer to applying Lemmas 4 and 3 to S_1 and S_2 by saying "we are done". We start with the properties of $S_1(v_i)$'s.

Claim C. *$S_1(v_i)$ is complete to $S_1(v_{i+2})$ and anti-complete to $S_1(v_{i+1})$. Further, if $S_1(v_i)$ and $S_1(v_{i+2})$ are both non-empty, then $|S_1(v_i)| \leq 3$ and $|S_1(v_{i+2})| \leq 3$.*

Proof of Claim C. Without loss of generality, it suffices to prove the claim for $S_1(v_0)$. Let $x \in S_1(v_0)$, $y \in S_1(v_1)$, and $z \in S_1(v_2)$. If $xz \notin E$, then $xv_0v_4v_3v_2y$ would be an induced P_6 in G. If $xy \in E$, then $v_0xyv_1v_0$ would be an induced C_4. Thus the first claim follows. Now suppose $|S_1(v_0)| \geq 4$. Then $S_1(v_0)$ contains two nonadjacent vertices x and x' since G is K_5-free. Now $xv_0x'zx$ would be an induced C_4. □

It follows from Claim C that we can pre-color all 1-vertices if at least four $S_1(v_i)$ are non-empty, or exactly three $S_1(v_i)$ are non-empty and three corresponding v_i's induce a $P_2 + P_1$ in G, or exactly two $S_1(v_i)$ are non-empty and two corresponding v_i's are non-adjacent. In all these cases we update the graph and we are done. The remaining cases are: (1) exactly one $S_1(v_i) \neq \emptyset$; (2) exactly two $S_1(v_i)$ are non-empty and two corresponding v_i's are adjacent; (3) exactly three $S_1(v_i)$ are non-empty and three corresponding v_i's induce a P_3.

Claim D. *$S_1(v_i)$ is anti-complete to all $S_2(v_j)$ for $j \neq i$. In addition, if both $S_1(v_i)$ and $S_1(v_{i+1})$ are non-empty, then $S_1(v_i)$ is also anti-complete to $S_2(v_i)$.*

Proof of Claim D. It suffices to prove the claim for $S_1(v_0)$. Let $x \in S_1(v_0)$, $y \in S_2(v_1)$ and $z \in S_2(v_2)$. If $xy \in E$, then xv_0v_4yx would induced a C_4. If $xz \in E$, then $v_1v_2v_3v_4zx$ would induce a P_6. By symmetry, the first part of the claim follows. Suppose now $S_1(v_0)$ and $S_1(v_1)$ are both non-empty. Let $x \in S_1(v_0)$, $y \in S_1(v_1)$ and $z \in S_2(v_0)$. If $xz \in E$, then $yv_1v_0xzv_3$ would induce a P_6 in G. □

It follows from Claim D that in the case (2) or (3) (we can pre-color two of three $S_1(v_i)$'s and update the graph) the 1-vertices that are not pre-colored are anti-complete to 2-vertices. Note also that in the case (2) the two non-empty $S_1(v_i)$'s are anti-complete to each other. So we are done in these two cases. Finally, we assume $S_1(v_0) \neq \emptyset$ and $S_1(v_i) = \emptyset$ for $i \neq 0$. If $|S_2(v_0)| \leq 2$, then we pre-color it, update the graph and we are done. So assume that $|S_2(v_0)| \geq 3$. If there is no edge between $S_1(v_0)$ and $S_2(v_0)$, then we are done by Claim D.

Hence, assume that there is at least one edge between $S_1(v_0)$ and $S_2(v_0)$.

Claim E. *$S_2(v_0)$ is a star.*

Proof of Claim E. Let $y \in S_2(v_0)$ be a neighbor of some vertex $x \in S_1(v_0)$. Suppose $y' \in S_2(v_0)$ is not adjacent to y. Consider $y'v_2yxv_0v_4$. Since G is P_6-free, we have $xy' \in E$ and thus $xyv_2y'x$ induces a C_4. Therefore, y is adjacent to any other vertex in $S_2(v_0)$. Thus $S_2(v_0)$ is a star since G is K_5-free. □

By Claim E we can pre-color $S_2(v_0)$ since there are exactly two such colorings. Finally we update the graph and we are done. Therefore, in any case we can decide in polynomial time if ϕ can be extended to S_1 and S_2.

Now we can assume that G is (C_4, C_5, P_6)-free. We proceed by appealing to the well-known lexicographical breadth first search (LeXBFS) to test whether or not G if chordal. If so, we can optimally color it in linear time. Otherwise, LeXBFS returns an induced cycle C of length great than 3. Since G is (C_4, C_5, P_6)-free, C must be an induced C_6. Using a similar argument above for $C = C_6$ we can tell if G is 4-colorable in polynomial time. This completes our proof. □

Time Complexity. The running time of our algorithm can be estimated as follows. To detect a K_5 or C_5 in the graph G it takes $O(n^5)$ time by brute force. The branch procedure reduces the 4-COLORING problem to 2-SAT, which can

be solved in linear time due to Aspvall et al. [2], and to 3-LIST COLORING. The running time of it is implicit in the analysis of the algorithm described by Broersma et al. [4], and can be easily verified to be $O(n^3)$. In the case where G contains an induced C_5, there are exactly one call for 2-SAT on S_2 and at most n calls for 3-LIST COLORING on $S_0 \cup S_1$ so that the running time is $O(n^5)$. When G does not contains an induced C_5, we obtain the same time bound $O(n^5)$ as LeXBFS runs in linear time. Consequently, the total running time of our algorithm is $O(n^5)$, and the bottleneck is to detect a K_5 or C_5.

4 Concluding Remarks

We have proved that 4-COLORING is NP-complete for P_7-free graphs, and that 5-COLORING is NP-complete for P_6-free graphs. These two results improve Theorems 3 and 4 obtained by Broersma et al. [4,6]. We have used a reduction from 3-SAT and establish a general framework. The construction and the proof are simpler than those in previous papers. As pointed out in Section 3, however, they do not apply to 4-COLORING P_6-free graphs. On the other hand, Golovach et al. [12] completed the dichotomy classification for 4-COLORING H-free graphs when H has at most five vertices. The classification states that 4-COLORING is polynomially solvable for H-free graphs when H is a linear forest and is NP-complete otherwise. Note that linear forests on at most five vertices are all induced subgraph of P_6. Thus, all the polynomial cases from [12] are for subclasses of P_6-free graphs. We conjecture that it can be decided in polynomial time if a P_6-free graph is 4-colorable.

Conjecture 1. 4-COLORING can be solved in polynomial time for P_6-free graphs.

As a first step towards to Conjecture 1, we have proved that it is true for (P_6, C_4)-free graphs, a subclass of P_6-free graphs. Our proof makes use of certain ideas of Le et al. [19] who proved that 4-COLORING is polynomially solvable for (P_5, C_5)-free graphs, and it also suggests new techniques that may be useful. Furthermore, Theorem 8 may be interesting in its own right. It suggests a new research direction, namely classifying the complexity of k-COLORING (P_t, C_l)-free graphs for every integer combination of k, l and t. Since k-COLORING is NP-complete for P_t-free graphs for even small k and t, say Theorems 5 and 6, it would be nice to know whether or not forbidding short induced cycles makes problem easier. In fact, Theorem 8 is a positive answer to the first non-trivial combination of (k, l, t). In contrast, one recent result of Golovach et al. [13] showed that 4-COLORING is NP-complete for (P_{164}, C_3)-free graphs. They also determined a lower bound $l(g)$ for any fixed $g \geq 3$ such that every $P_{l(g)}$-free graph with girth at least g is 3-colorable. Note that the girth condition implies the absence of all induced cycles of length from 3 to $g-1$ in the graph. Therefore, the last result can be viewed as an answer to a restricted version of the problem we have formulated.

Acknowledgement. The author is grateful to his supervisor Pavol Hell for many useful conversations related to these results and helpful suggestions for improving the presentation of the paper.

Note added in proof. We have recently heard from Daniel Paulusma that they proved a more general result than Theorem 8. In particular, they showed that k-COLORING can be solved in linear time for $(K_{r,s}, P_t)$-free graphs for any fixed integer k, r, s, t; however, their linear time algorithm has huge constants as it relies strongly on a recent result of Atminas [1] involving Ramsey number and treewidth algorithm. Our algorithm which runs in $O(n^5)$ time may be more practical for up to a fairly large input size n.

References

1. Atminas, A., Lozin, V.V., Razgon, I.: Linear time algorithm for computing a small biclique in graphs without long induced path. In: Proreedings of SWAT 2012, LNCS vol. 7357 (2012) 142-152.
2. Aspvall, B., Plass, M.F., Tarjan, R.E.: Linear-time algorithm for testing the truth of certain quantified boolean formulas. Information Processing Letters 8, 121-123 (1979).
3. Bondy, J.A., Murty, U.S.R.: Graph Theory. In: Springer Graduate Texts in Mathematics, vol. 244 (2008).
4. Broersma, H.J., Fomin, F.V., Golovach, P.A., Paulusma, D.: Three complexity results on coloring P_k-free graphs. European Journal of Combinatorics, 2012 (in press).
5. Broersma, H.J., Golovach, P.A., Paulusma, D., Song, J.: Determining the chromatic number of triangle-free $2P_3$-free graphs in polynomial time. Theoret. Comput. Sci. 423, 1–10 (2012).
6. Broersma, H.J., Golovach, P.A., Paulusma, D., Song, J.: Updating the complexity status of coloring graphs without a fixed induced learn forest. Theoret. Comput. Sci. 414, 9–19 (2012).
7. Chudnovsky, M., Cornuéjols, G., Liu, X., Seymour, P., Vušković, K.: Recognizing Berge graphs. Combinatorica 25, 143–187 (2005).
8. Chudnovsky, M., Robertson, N., Seymour, P., Thomas, R.: The strong perfect graph theorem. Annals of Mathematics 64, 51–229 (2006).
9. Dabrowski, K., Lozin, V.V., Raman, R., Ries, B.: Colouring vertices of triangle-free graphs without forests. Discrete Math. 312, 1372–1385 (2012).
10. Edwards, K.: The complexity of coloring problems on dense graphs. Theoret. Comput. Sci. 43, 337–343 (1986).
11. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman San Faranciso, (1979).
12. Golovach, P.A., Paulusma, D., Song, J.: 4-coloring H-free graphs when H is small. Discrete Applied Mathematics, 2012 (in press).
13. Golovach, P.A., Paulusma, D., Song, J.: Coloring graphs without short cycles and long induced paths. In: Proceedings of FCT 2011, in: LNCS, vol. 6914, 2011, pp. 193-204.
14. Grötschel, M., Lovász, L., Schrijver, A.: Polynomial algorithms for perfect graphs. Ann. Discrete Math. 21, 325–356 (1984). Topics on Perfect Graphs.

15. Hoàng, C.T., Kamiński, M., Lozin, V.V., Sawada, J., Shu, X.: Deciding k-colorability of P_5-free graphs in polynomial time. Algorithmica 57, 74–81 (2010).
16. Holyer, I.: The NP-completeness of edge coloring. SIAM J. Comput. 10, 718–720 (1981).
17. Kamiński, M., Lozin, V.V.: Coloring edges and vertices of graphs without short or long cycles. Contrib. Discrete. Mah. 2, 61–66 (2007).
18. Král, D., Kratochvíl, J., Tuza, Zs., Woeginger, G.J.: Complexity of coloring graphs without forbidden induced subgraphs. In: Proceedings of WG 2001, in: LNCS, vol. 2204, 2001, pp. 254-262.
19. Le, V.B., Randerath, B., Schiermeyer, I.: On the complexity of 4-coloring graphs without long induced paths. Theoret. Comput. Sci. 389, 330–335 (2007).
20. Leven, D., Galil, Z.: NP-completeness of finding the chromatic index of regular graphs. J. Algorithm 4, 35–44 (1983).
21. Randerath, B., Schiermeyer, I.: 3-Colorability $\in$ P for P_6-free graphs. Discrete Appl. Math. 136, 299–313 (2004).
22. Randerath, B., Schiermeyer, I.: Vertex colouring and fibidden subgraphs-a survey. Graphs Combin. 20, 1–40 (2004).
23. Tuza, Zs.: Graph colorings with local restrictions-a survey. Discuss. Math. Graph Theory 17, 161–228 (1997).
24. Woeginger, G.J., Sgall, J.: The complexity of coloring graphs without long induced paths. Acta Cybernet. 15, 107–117 (2001).